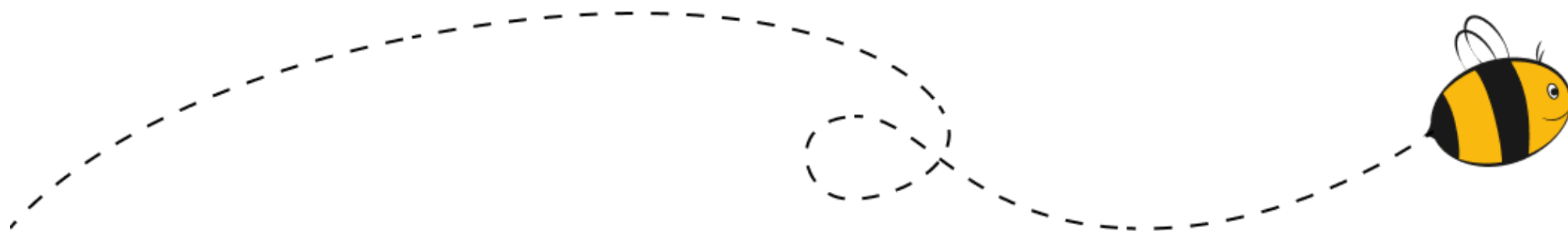
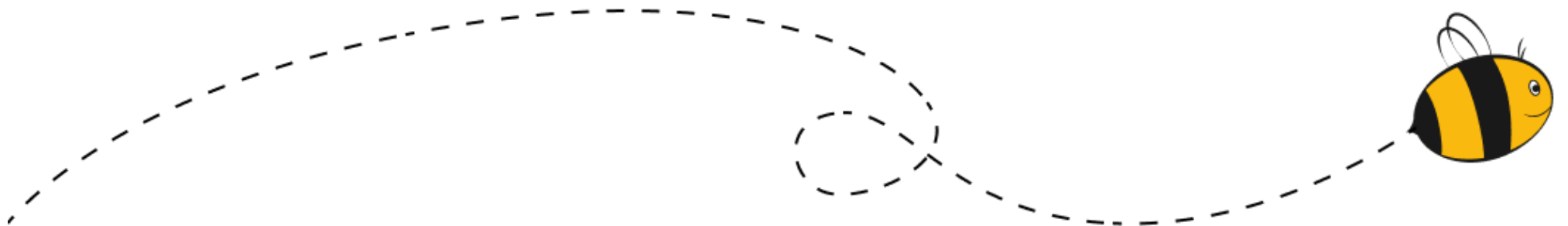
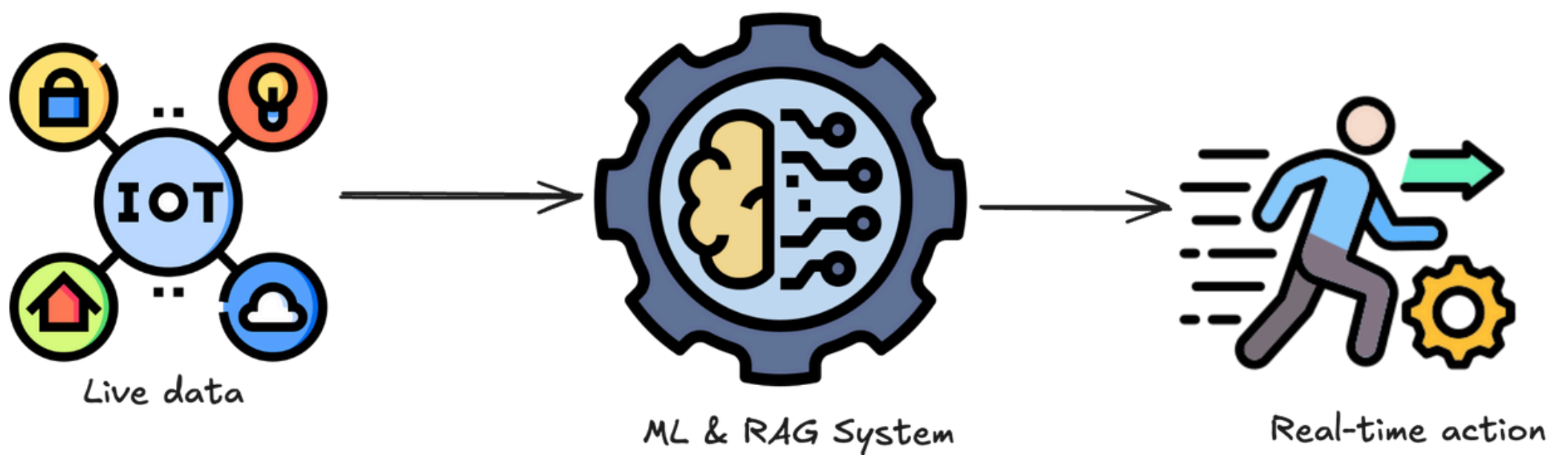


# Bring streaming to AI: Introducing Bytewax connectors



# Why streaming for AI

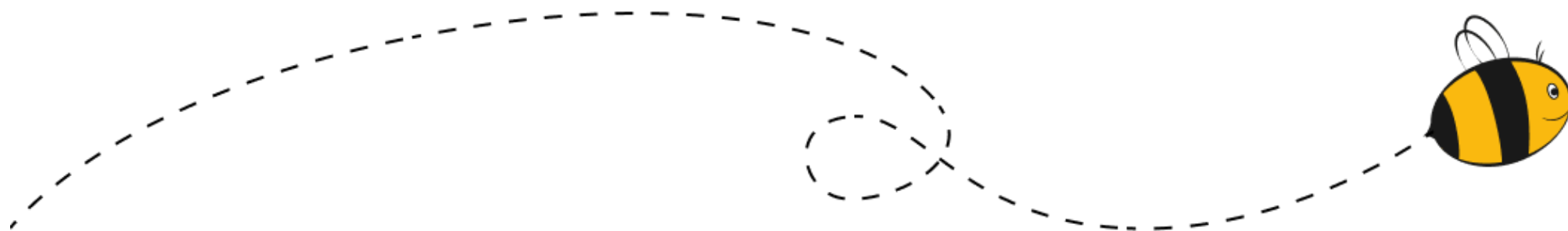


# Why streaming for AI

**Process data as it arrives**

**Gain insights in real time**

**Execute actions with latest data**

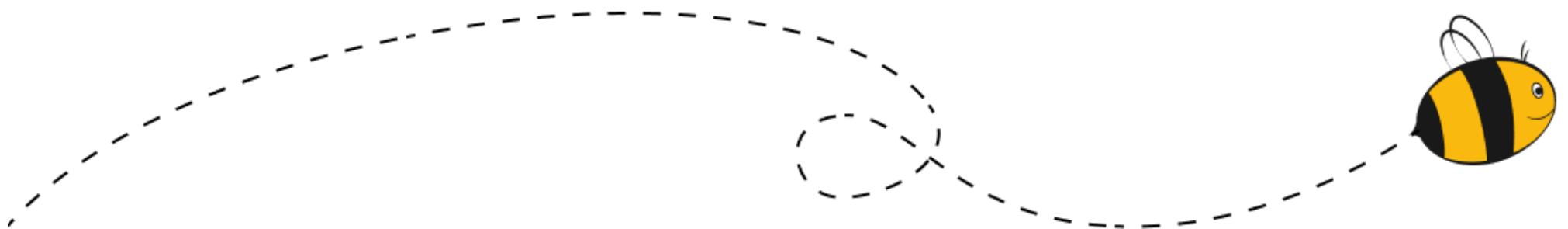


# Challenges in real-time data processing

**Traditional batch processing falls short for real-time AI applications.**

**Building in-house connectors and pipelines is time-intensive and error-prone.**

**Need for scalable, streamlined, reusable solutions**

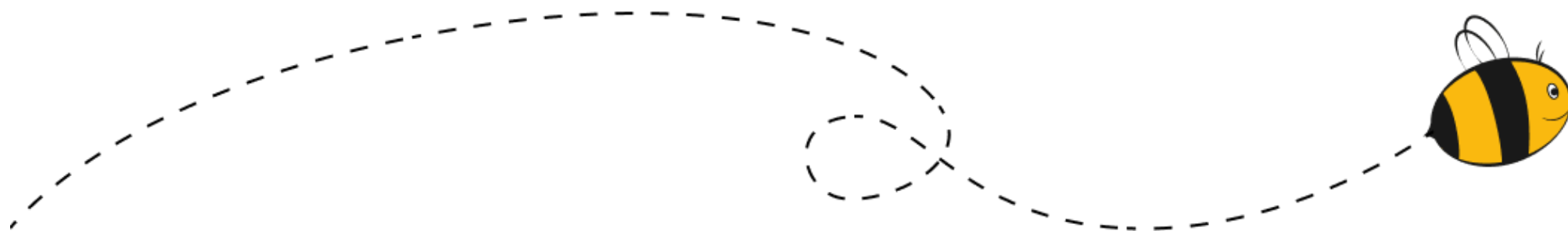


# Introducing Bytewax

**Python-native framework for real-time  
stream processing**

**Open source, it combines the performance of  
Rust with ease of use of Python**

**500K+ downloads on PyPi**



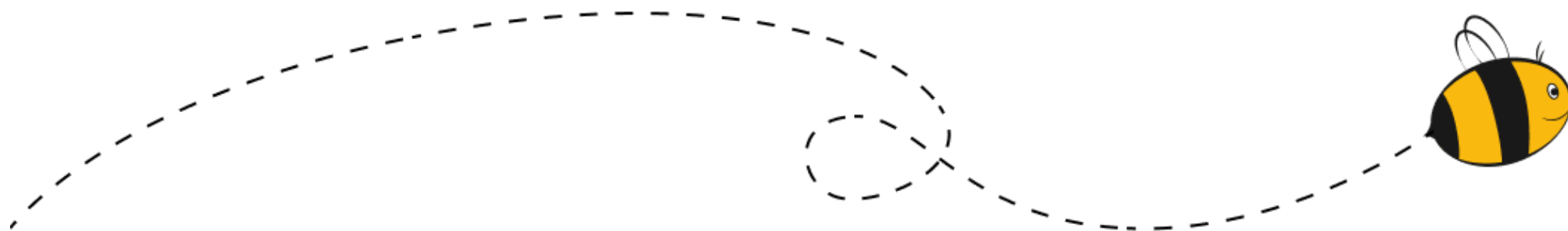
# Easy installation, Python native

```
> pip install bytewax
```

```
from bytewax import operators as op
from bytewax.connectors.kafka import operators as kop
from bytewax.dataflow import Dataflow

BROKERS = ["localhost:19092"]
IN_TOPICS = ["in_topic"]
OUT_TOPIC = "out_topic"

flow = Dataflow("kafka_in_out")
kinp = kop.input("inp", flow, brokers=BROKERS, topics=IN_TOPICS)
op.inspect("inspect-errors", kinp.errs)
op.inspect("inspect-oks", kinp.oks)
kop.output("out1", kinp.oks, brokers=BROKERS, topic=OUT_TOPIC)
```

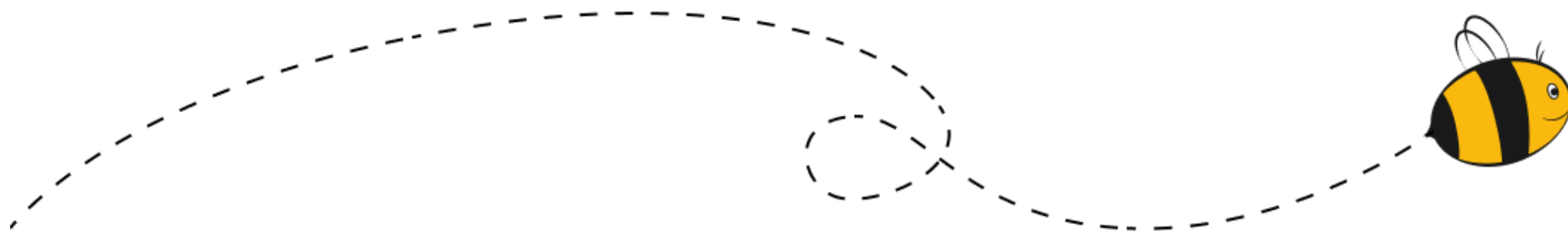


# Bytewax connectors: the gateway to Streaming AI

**Connectors as the link between real-time data sources and AI pipelines.**

**Source connectors ingest data from APIs, IoT devices, databases, etc.**

**Sink connectors send processed data to analytics tools, databases, or AI models.**



# Bytewax Dataflow

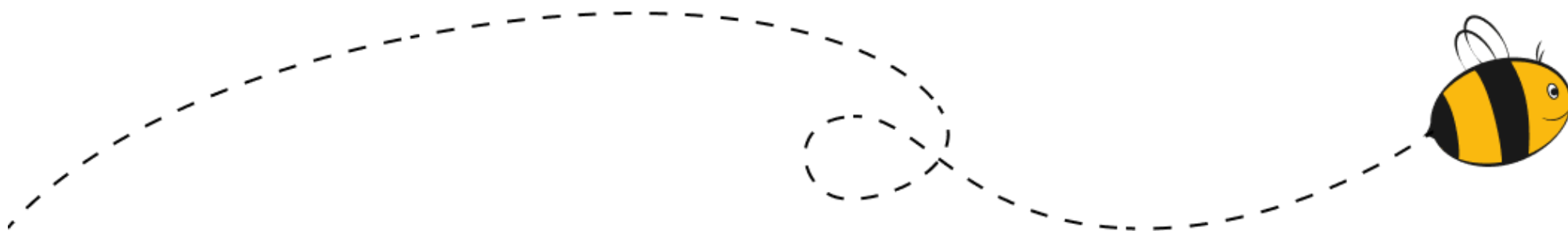
**Source connector**



**AI Pipeline**

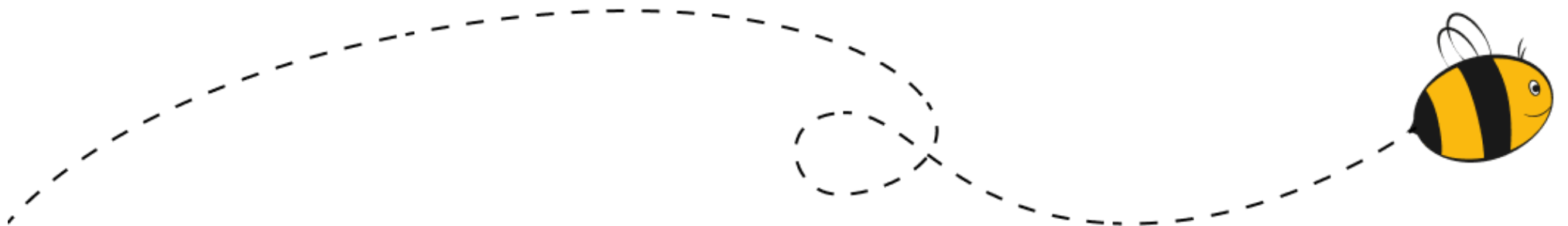
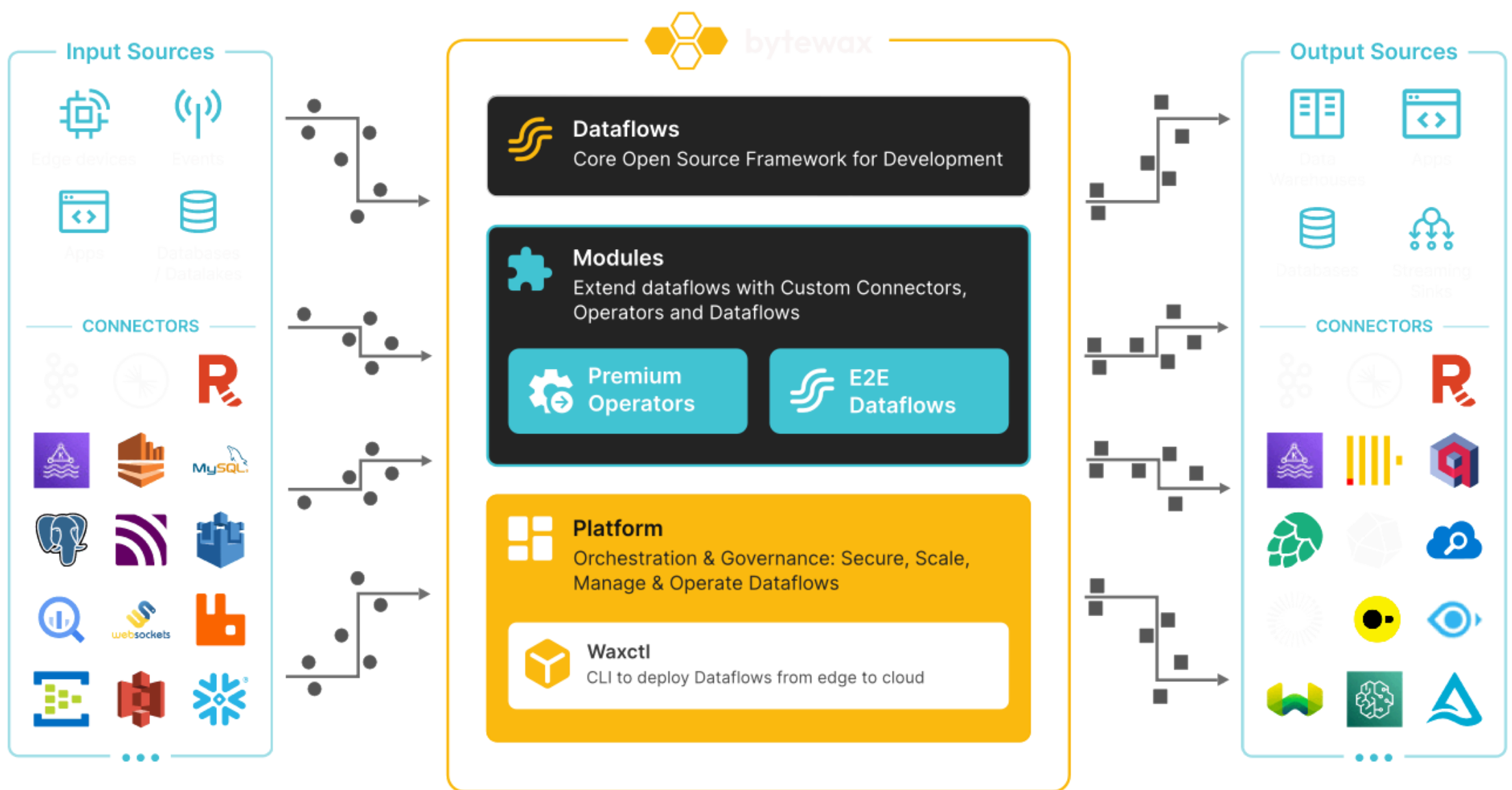


**Sink connector**





# Build end to end real-time end to end applications

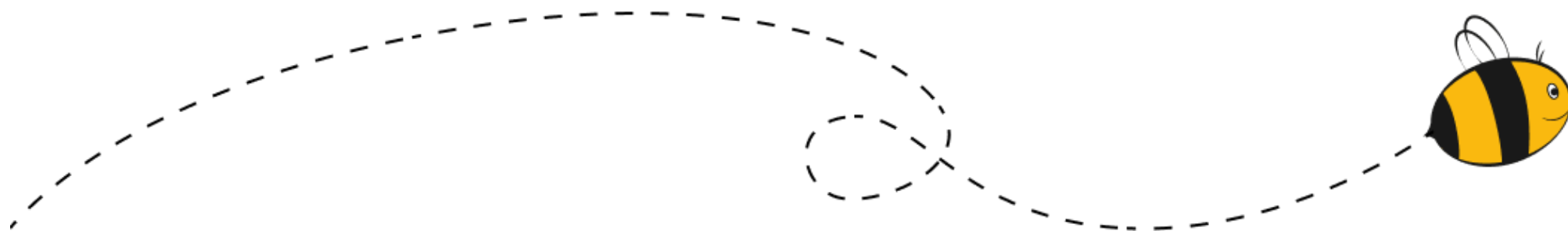


# About the connectors

**Install Bytewax modules via pip**

**Licensing options: open-source, commercial and premium**

**Community support through Slack and documentation**



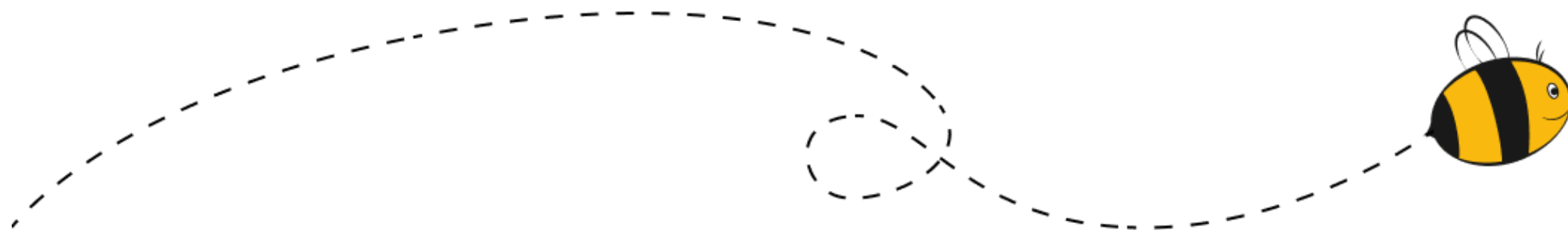
# Real-Time AI Use Cases with Bytewax

**Streaming data from IoT sensors for predictive maintenance**

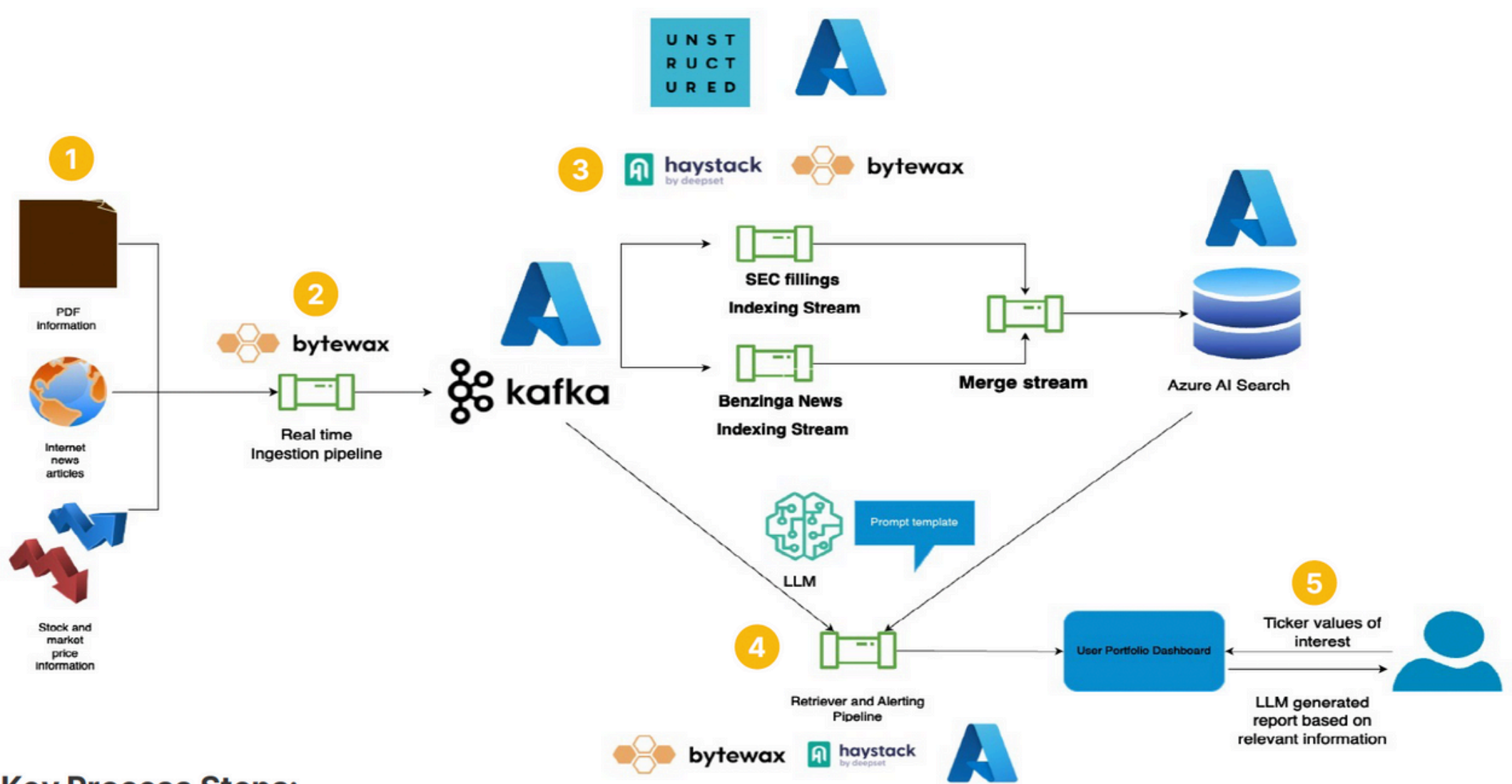
**Real-time recommendation systems in e-commerce**

**Adaptive machine learning with live feedback loops**

**Bytewax empowers these use cases with modular, extensible connectors.**

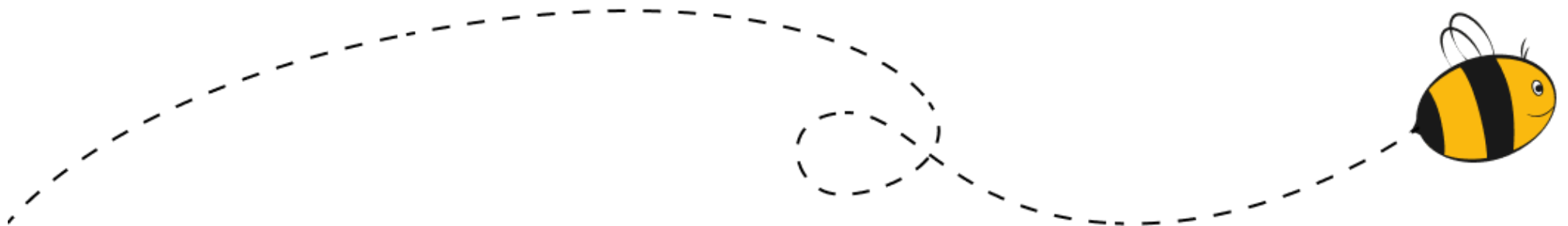


# Example scenarios: RAG workflow

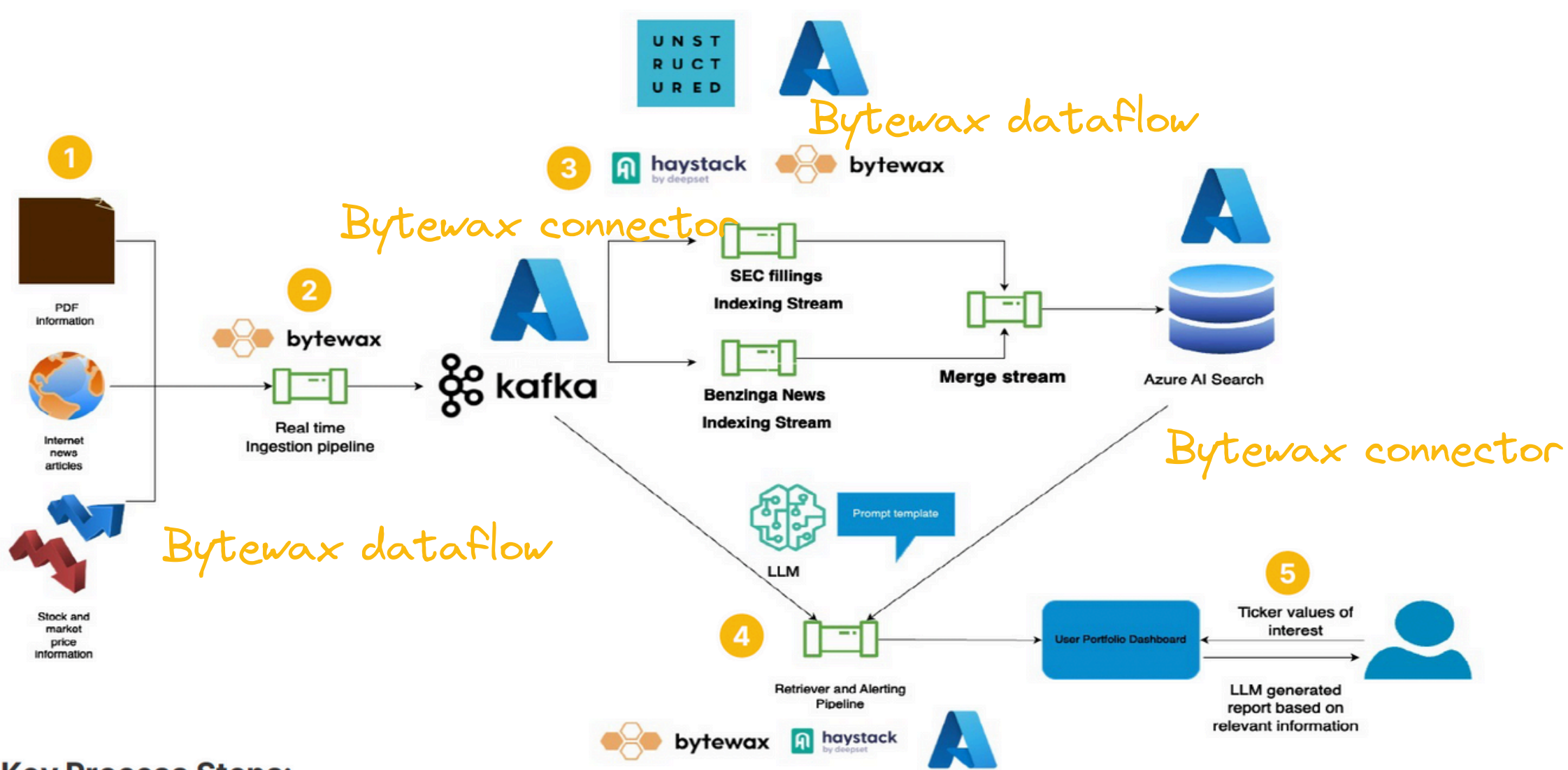


## Key Process Steps:

- 1** Data extraction
- 3** Data indexing
- 5** Response generation with an LLM
- 2** Data wrangling and pre-processing
- 4** Data retrieval

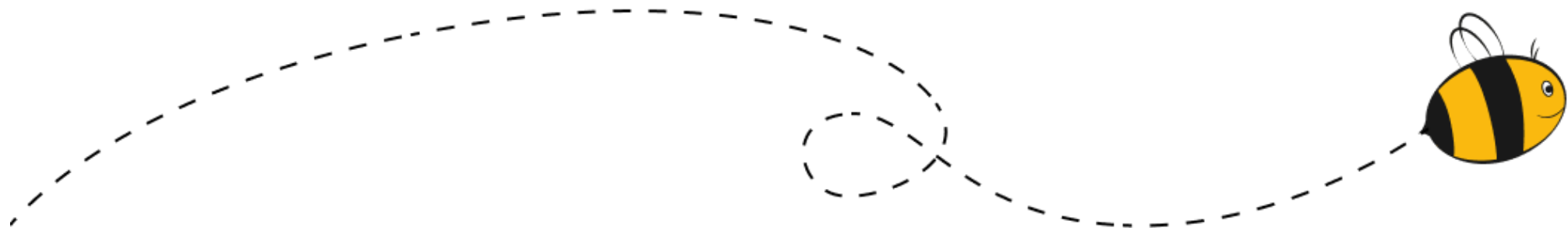


# Example scenarios: RAG workflow



## Key Process Steps:

- 1 Data extraction
- 2 Data wrangling and pre-processing
- 3 Data indexing
- 4 Data retrieval
- 5 Response generation with an LLM





# Azure AI Search Sink connector

CI passing pypi v0.1.2 user guide



## bytewax-azure-ai-search

Custom sink for [Azure AI Search](#) vector database for real time indexing.

bytewax-azure-ai-search is commercially licensed with publicly available source code. Please see the full details in [LICENSE](#).

## Installation and import sample

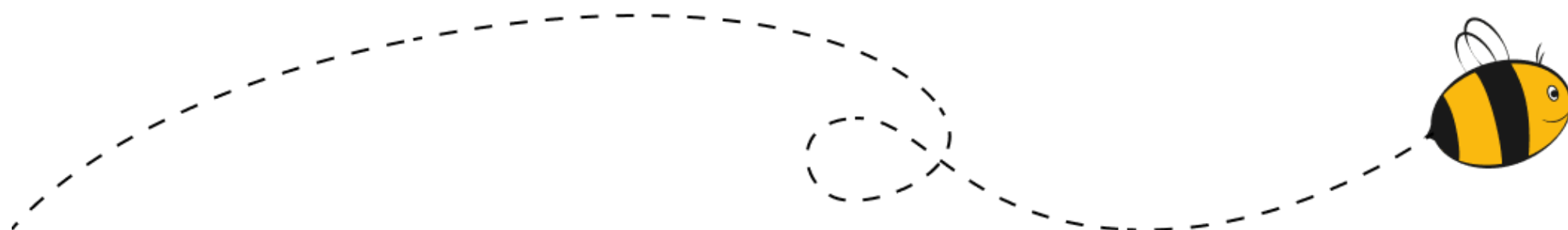
To install you can run

```
pip install bytewax-azure-ai-search
```

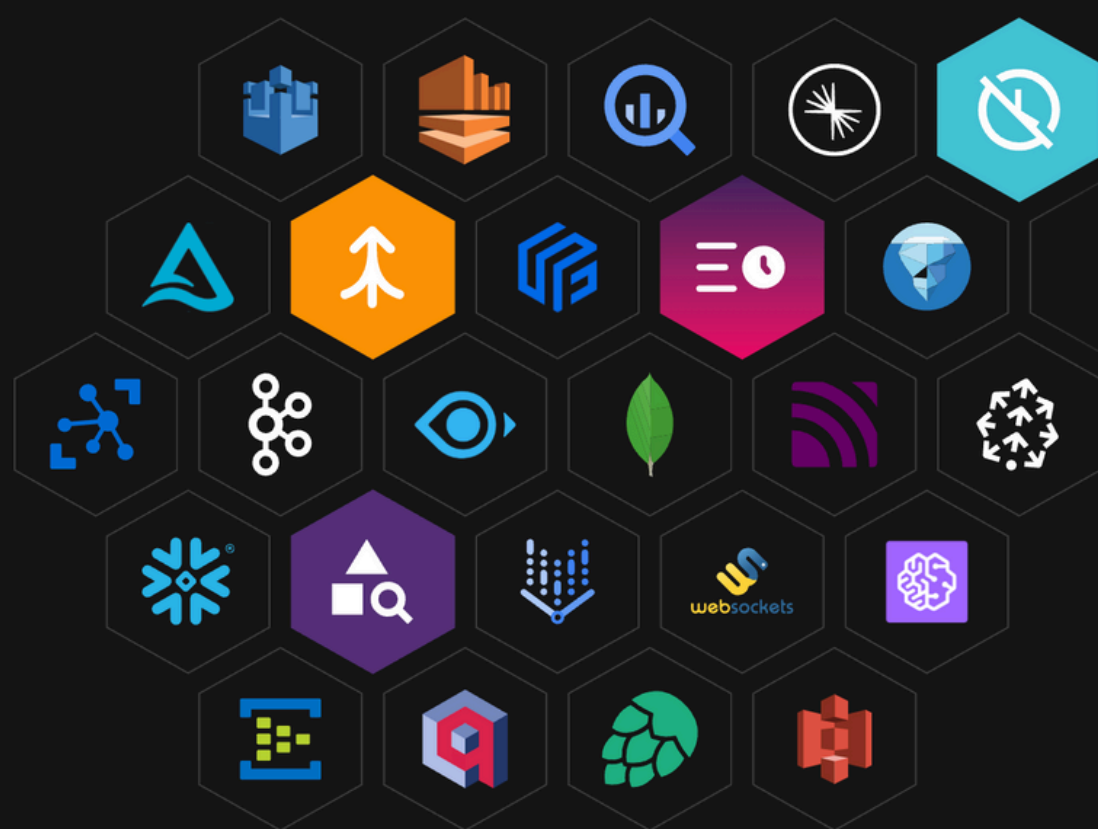


Then import

```
from bytewax.azure_ai_search import AzureSearchSink
```



# Learn about current connectors offered



## MODULES

### Extend Bytewax with connectors, operators, and E2E dataflows

Our Module Hub extends the open source dataflows framework with pre-built connectors to countless sources and sinks, advanced operators, and E2E dataflows

EXPLORE MODULES 

<https://bytewax.io/blog/bytewax-open-source-modules>

